# MAT2: Enhanced Machine Aided Translation System

**K. Narayana Murthy**
Department of Computer and Information Sciences,
University of Hyderabad,
Hyderabad, 500 046,
INDIA
email: knmcs@uohyd.ernet.in

## Abstract

This paper describes the recent developments and enhancements to the MAT machine aided translation system that was first developed by the author for the Government of Karnataka in the late nineties. MAT was used for translating English texts into Kannada, the official language of Karnataka. MAT is a parser based translation aid, ideally suited for translating between positional languages like English and Indian languages, which are characterized by a relatively free word order and a very rich system of morphology. MAT is based on the UCSG theory of syntactic analysis developed by the author. In this paper a brief sketch of the MAT system is presented first. Then the enhancements being made to the UCSG parsing system and the new features being incorporated into the current version of translation system MAT2 are described.

**Keywords: Machine Translation, Parser Based Translation**

## 1 Introduction

It is now fairly clear that fully automatic high quality translation is difficult to realize in practice. MAT is a machine assisted translation system that provides for a full spectrum of possibilities - from fully automatic generation of raw translations suitable for manual post editing, through semi-automatic translation to almost fully manual translation using the facilities provided by the system. The basic idea is to make the best of both the human and machine capabilities to achieve good translations with minimum time and effort. Apart from a very powerful post editing tool, MAT also comes with dictionaries, a thesaurus, morphological analyzer/generator and several other useful tools. MAT2 carries this idea of man-machine synergy further and purports to combine human judgement with machine learning techniques based on corpora.

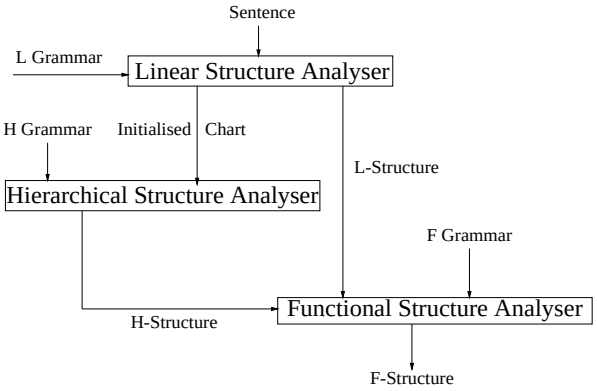
MAT is a parser based translation system. Each sentence in the source text is parsed syntactically before translating. This makes MAT suitable for translating between languages that show a significant variation in sentence structures. MAT was developed especially for translating between English and Indian languages but it can be applied to other languages too (Murthy1996b; Murthy1999b).

The next section describes briefly the UCSG parsing system that is used in MAT. The following section deals with the translation and post processing modules. Then we take up the developments and enhancements being made to the UCSG parsing system and the translation scheme in the new MAT2 system.

## 2 Parsing in UCSG

This section describes briefly the UCSG parsing system that is used in MAT. Given here is only a brief sketch and for further details and theoretical issues, the reader may refer to (Murthy1996a; Murthy1997a; Murthy1997b). In UCSG we find that the three primary kinds of structure inherent in human languages namely, linear, hierarchical and functional structures, lend themselves naturally for analysis by three independent modules. UCSG proposes three levels of representation called *L-Structure, H-Structure* and *F-Structure* along with the corresponding components of the grammar and parser. In UCSG we *divide and*

*conquer*. We apply the least powerful kind of grammar for each subtask, making the parsing process computationally efficient on the whole. Also, this modularization makes it easy to write grammars. The division of labour into these three separate modules, especially the introduction of an independent H-Structure is the highlight of UCSG. We have shown that this modularization makes UCSG efficient and robust for analyzing both positional and free word order languages on an equal footing. The following block diagram gives the overall architecture of UCSG:




The L-Module picks up one sentence at a time, breaks it up into words and looks up each word in the dictionary, calling the morphological analyzer if and where required. It outputs all potential word groups (or 'phrases') in the sentence. The L grammar is a Nondeterministic Finite State Automaton. It has been shown that all potential word groups in a sentence can be recognized in a single scan of the sentence in linear time. All subsequent levels of processing look only at whole word groups, never the individual words. Thus the effective length of the sentence is reduced and the rest of the parsing process becomes so much more efficient.

Functional structure - the thematic roles or 'functions' taken up by various noun groups, is essentially local to a clause. Each clause has its own verb and the associated subject, object, and so on. It is a fundamental principle of language structure that functional units of a clause respect the clause boundaries. Although exceptions exist, as a rule functional units of one clause do not cross clause boundaries. UCSG attempts to analyze the hierarchical structure of clauses and to determine clause boundaries before attempting to analyze the functional structure of each clause. UCSG also shows (Murthy1996a) that it is possible to determine clause structure of complex sentences - the number and nature of clauses and their inter-relationships - before and without applying any of the functional structure constraints. In fact we can determine the clause boundaries also, albeit only partially.

Knowing the clause boundaries has great significance to parsing since the functional structure of a clause is essentially local to that clause. Thus we can *work from whole to part* rather than from left to right. In UCSG, the clause structure is determined in the H-module. This module takes the sequence of only verb groups and certain clause boundary markers called sentinels as input and produces a parse tree - the clause structure tree using a very small number of *Context Free Grammar* (CFG) rules. UCSG employs an *active chart parsing algorithm* for the H-level. Although the complexity of parsing with general CFGs is $O(n^3)$, here 'n', the input length is much smaller than the length of the sentence. Further, the number of grammar rules required is also very small. Thus the H-module itself is very efficient.

Then the F-module assigns functional roles such as 'subject' and 'object' to various word groups in each clause in the sentence. To do this, we bring to bear simultaneously the top-down constraints in the form of expectations of the verb groups and the bottom-up constraints in the form of available word groups and their grammatical features. All possible functional role assignments that satisfy all the constraints are noted. There is a rating system that rates the role assignments based on a number of hard and/or soft constraints.

After making all possible role assignments, a *best-first search* algorithm picks up the best possible set of role assignments. This scheme has several merits. Firstly, we can hope to get the best parses first without compromising on the ability of the system to generate all the potential parses. If the first parse, or one of the top few parses is found acceptable, the others are not even generated. Also, a

variety of statistical rating schemes can be incorporated easily for better ranking. Finally, this technique gives robustness to the parsing system - even ungrammatical or incomplete sentences can be parsed, albeit with a penalty.

The F-module takes up each clause, starting from the matrix clause and working inwards, down the clause structure tree, passing on expectations and information about the inter-clause dependencies if any. This makes the functional structure analysis of a clause essentially independent of all other clauses in the sentence, thereby reducing the computational complexity of parsing significantly. The H-module has made it possible to assign $r$ roles to $r$ word groups $c$ times for a sentence with $c$ clauses rather than attempting to assign $c * r$ roles to $c * r$ word groups all at once. The additional effort in analyzing clause structure is much more than compensated for by this. Parsing in UCSG is thus highly efficient, making it practicable for many applications including machine assisted translation. UCSG has also been used in other areas such as interpreting metaphors (Varma1996).

## 3 The Translator:

In MAT, like the parser, the translator also works from whole to part, rather than from left to right. After a sentence has been parsed by the UCSG parser, we would know the number, type and inter-relationships amongst the various clauses in the sentence and the word groups that take on various functional roles in each of these clauses. Keeping this structure of the sentence in mind, a suitable structure for the equivalent sentence in the target language is first developed. Where it is not feasible to make more or less direct transfer of structure, sentence transformation rules can be called to restructure the parse structure so that it becomes amenable for translation. This makes it possible to deal with languages with vastly different sentence structures.

After having fixed the overall structure of the target language sentence, the individual clauses are mapped. Finally, the various word groups are mapped. In each word group, the head and the modifiers are identified and transfered, keeping in mind the L-grammar of the target language. For each word, a suitable target language equivalent is obtained from the bilingual dictionary. The MAT system provides for incorporating syntactic and some simple kinds of semantic constraints in the bilingual lexicon for word sense disambiguation. For example, the word 'rise' is mapped differently into Kannada in the following three sentences:

| | |
|---|---|
| I *rose* | na:nu *eddenu* |
| The moon *rose* | caMdra *huTTitu* |
| The prices are *rising* | belegaLu *heccuttive* |

Kannada, the target language in the current experiments, is morphologically very rich. Words in Kannada often take as many as 6 suffixes. In many cases, a whole word group in English translates into a single highly inflected word in Kannada. The MAT system includes a morphological analyzer/generator for Kannada (Murthy1999a; Sridhar1990). Using the Kannada morphological generator, appropriate word forms are generated in each case. For example, in Kannada the subject has to agree with the verb not only in number but also in gender. So gender feature is extracted from the subject of the English source sentence and used in the morphological generation of the Kannada verb so that, for example, 'she came' becomes 'avaLu baMdaLu' and 'he came' becomes 'avanu baMdanu' in Kannada. Finally, the target language sentence is generated by placing the clauses and the word groups in appropriate linear order, according to the constraints of the target language grammar.

In case a complete parse with satisfactory rating was not obtained for whatever reason, the word groups with or without the functional roles assigned to them are available for semi-automatic translation. The user picks up parts of the sentence, interactively calls the translator module to translate the part and finally assists the machine in assembling the target language sentence.

MAT can be run in one of the three modes called *non-interactive*, *interactive* and *custom* modes. A time limit can be specified to instruct

the system to skip a sentence if it is taking too much time. Performance of the system is displayed continuously and a summary and a histogram are displayed at the end.

### 3.1 The Post Editing Tool:

The post-editing tool displays the source text and the corresponding translated target language text one sentence at a time. Using the tool, the human translator can move around pieces of text easily. He can also delete, insert and edit words at will. More significantly, he can call the thesaurus on line and substitute selected words by their equivalents. *Morphological analysis and generation are done on the fly* so that the correct word forms are substituted automatically. Substitution can be called on both the target and source language words. One can also directly invoke the morphological analyzer, look at and modify the feature list and then re-generate a new word form. Further, it is also possible to call the translator to translate selected parts of a text. These unique advanced features make it possible to translate full sized English texts with a minimum of effort. As a last resort the user can manually retype the correct translation. Machine aided translation permits high quality translations to be obtained irrespective of the complexity of sentence structures employed and the inherent limitations of the parsing technology. Overall, the post editing tool makes the life of the translator so much easier and gives significant time savings too.

### 3.2 Other Tools in MAT

There are separate monolingual dictionaries for English and Kannada as also a bilingual English-Kannada dictionary. A unique feature of this system is that the bilingual dictionary can be used as a kind of thesaurus too. You can get all *related words* for a given word.

There is also a morphological analyzer cum generator for Kannada. Kannada words can be analyzed for their internal structure. Specific word forms can also be generated from a given root word. Further, it is possible to get a full paradigm, that is, a systematic listing of all the forms of a given word. Both noun and verb morphology are included. The current version is limited to inflectional morphology - there is neither any derivational component nor rules for inter-word saMdhi nor for compounding.

A 'Pre-Scan' utility is provided to obtain a list of words from the given text whose entries are not found in the dictionary. All the required entries will be indicated in the appropriate structure in a file called 'Custom.dat'. The 'Custom.dat' file is read each time the MAT system is run. So all changes and enhancements made in this file will be effective in all future runs. However, in case you wish to incorporate the entries here directly into the main dictionary, an 'Update-Dictionary' tool is provided.

## 4 MAT2:

The MAT system version 1.0 was completed in January 1999 and has been tested on several budget speech texts. MAT can parse and translate about 40 to 60 percent of sentences fully automatically. The translations so produced by the machine are often in more or less acceptable form. Little or no post editing may be required in such cases. Changes required are mostly stylistic in nature. Primary meaning is preserved. However, some sentences may need substantial editing and in a few rare cases, the outputs may have to be rewritten completely. Where automatic translation fails, semi-automatic translation is possible - user selects parts of the sentence, calls the translator to produce translations for the parts, and finally assembles the parts into the complete target language sentence. In all cases, high quality translations can be obtained quickly using the post editing tool. Overall, the system can bring in substantial time savings.

The MAT system, like most other Automatic Translation systems, is far from ideal. An ideal MT system needs to have access to world knowledge and common sense based decision making. How would the system differentiate between 'mothers with babies above 4 years' from 'mothers with babies above 40 years'? A lot of progress has been made in recent past in dealing with Word Sense Disambiguation

(WSD). Yet WSD and lexical choice remain difficult challenges. Machines still have a lot of difficulty dealing with new words and novel expressions. Availability of large corpora has made new statistical techniques promising. Yet corpus based statistical methodologies cannot solve all the problems. Therefore the best bet continues to be a well designed man-machine synergy. Pre-editing is rarely an acceptable proposition. Post-editing needs should also be reduced. User acceptance cannot be achieved even if there is a single catastrophic case where the meaning or the intention has been completely distorted. In MAT2 we propose a machine aided translation architecture that purports to build a synergistic man-machine system that exploits the best of human and machine capabilities. MAT2 allows non-interactive and custom modes like its earlier version but carries the interactive mode further. Below we give the essence of the new approach.

MAT2 uses the new, improved UCSG parsing system. The English dictionary now has about 60,000 entries carefully selected by analyzing large corpora. The dictionary gives frequency information apart from syntactic categories and features. The new L module uses an improved Finite State device. The chunks so recognized are rated and ranked using statistical methods based on Mutual Information. It is easier to achieve robustness for shallow parsing than for full parsing. In MAT2 it is possible to do only the L level parsing and go ahead with translation. Of course full parsing can also be resorted to wherever appropriate. After L level parsing, the source language word groups are translated into target language groups. Keeping target language sentence structure in mind, the translated chunks are presented in the right order in dynamically generated menus for the interactive user to choose from. Alternatives in the menu are presented in order of their rank. Each step by the user in selecting an item from the menu restricts further choices as per a variety of compatibility constraints. Machine learning techniques are combined with user's choices to deal with problems such as WSD and lexical choice. Thus the system 'learns' from the user as well as from its own experience, and quickly adapts to a new domain. There is no absolute guarantee that the machines always guesses right but it is possible to achieve a high degree of correct guesses.

Thus while the real hard problems of world knowledge and common sense, new words and novel expressions, gaps in grammar and ungrammatical input, are actually taken care of by the human user, the machine does its bit through syntactic parsing using both linguistic and statistical techniques, adapts quickly to current needs, and offers a set of choices for the user to choose from. Thus we avoid catastrophic mistakes in translation and quality of translation is guaranteed. In the process, annotated bilingual corpora are gradually developed, paving the way for better systems in the future.

## 5 Conclusions:

We have described briefly the MAT2 system for parser based semi-automatic translation that combines traditional linguistic approach with corpus based statistical processing technologies to achieve high quality translations with minimum effort. The MAT2 system is still under development. Experiments with human translators to obtain quantitative measures of system effectiveness will be conducted once the system is ready.

## References


K Narayana Murthy. *Universal Clause Structure Grammar*. PhD thesis, Dept. of Computer and Information Sciences, University of Hyderabad, 1996a.

K Narayana Murthy. Parsing Telugu in the UCSG Formalism. In *Proceedings of the Indian Congress on Knowledge and Language*, volume 2, pages 1–16, 1996b.

K Narayana Murthy and A Sivasankara Reddy. Universal Clause Structure Grammar. *Computer Science and Informatics*, 27(1):26–38, 1997a.

K Narayana Murthy. Universal Clause Structure Grammar and the Syntax of Relatively Free Word Order Languages. *South Asian Language Review*, VII(1):47–64, 1997b.

K Narayana Murthy. A Network and Process Model for Morphological Analysis/Generation. In *Proceedings of the Second International Conference on South Asian Languages*, 1999a.

K Narayana Murthy. MAT: A Machine Assisted Translation System. In *Proceedings of the NLPRS-99 Fifth Natural Language Processing Pacific Rim Symposium, Beijing, China, Nov 5-7*, 1999b.

Vasudev Varma and A Sivasankara Reddy. Knowledge based metaphor interpretation. *Knowledge-Based Systems*, 9:339–342, 1996.

S N Sridhar. *Kannada*. Routledge, 1990.